

# Integrate jenkins with sonarqube

The screenshot shows the AWS CloudWatch Instances interface. At the top, there's a header with 'Instances (2) Info' and various navigation buttons like 'Connect', 'Instance state', 'Actions', and 'Launch instances'. Below the header is a search bar and a table listing two instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Z
jenkins	i-02843da0d4661ac63	Running	c7i-flex.large	Initializing	<a href="#">View alarms +</a>	us-east-1a
sonarcubek	i-0c9016dbe603bd047	Running	t2.micro	2/2 checks passed	<a href="#">View alarms +</a>	us-east-1a

Below the table, there's a section titled 'Select an instance' with a dropdown menu. The 'jenkins' instance is selected, and its details are shown in a large black box:

```
System load: 0.08      Temperature: -273.1 C
Usage of /: 25.6% of 6.71GB Processes: 147
Memory usage: 5%       Users logged in: 0
Swap usage: 0%          IPv4 address for enp39s0: 172.31.85.65

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-85-65:~$
```

At the bottom of this box, it says 'i-02843da0d4661ac63 (jenkins)' and 'PublicIPs: 54.166.22.14 PrivateIPs: 172.31.85.65'.

The 'sonarcubek' instance is also listed below with its own details in a black box:

```
System load: 0.0      Processes: 106
Usage of /: 25.6% of 6.71GB Users logged in: 0
Memory usage: 20%     IPv4 address for enX0: 172.31.87.88
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-87-88:~$
```

At the bottom of this box, it says 'i-0c9016dbe603bd047 (sonarcubek)' and 'PublicIPs: 3.92.199.70 PrivateIPs: 172.31.87.88'.

The screenshot shows the GitHub repository page for 'Deployment-script'. The repository is public and was forked from 'akshu20791/Deployment-script'. The main branch is up-to-date with 'akshu20791/Deployment-script:main'. The commit history for the main branch includes:

- Update efs-ubuntu.sh by akshu20791 (0e796f1, 3 weeks ago)
- Update docker.sh by docker (9 months ago)
- Update efs-ubuntu.sh by efs-ubuntu (3 weeks ago)
- Rename grafana to grafana.sh by grafana (5 months ago)
- Update jenkins.sh by jenkins (last year)
- Update k8s-master.sh by k8s-master (last year)
- Update k8s-nodes.sh by k8s-nodes (last year)
- Update readme-k8s by readme-k8s (11 months ago)
- Create trivy.sh by trivy (9 months ago)

The repository has 0 stars, 0 forks, and 0 watching. It has 25 commits in total.

<https://github.com/akshu20791/Deployment-script>

Go To Jenkins  
Raw

The screenshot shows a browser window displaying the raw content of the Jenkins setup script. The URL is <https://raw.githubusercontent.com/akshu20791/Deployment-script/main/jenkins.sh>. The script content is as follows:

```
#!/bin/bash
# USE UBUNTU20.04 - INSTANCE: 2GB RAM + 2VCPU MIN - WILL ONLY WORK
sudo apt update -y
sudo apt install openjdk-17-jdk -y
sudo apt update -y
sudo apt install maven -y
curl -fssl https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee \
    /usr/share/keyrings/jenkins-keyring.asc > /dev/null
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
    https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
    /etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt update -y
sudo apt install jenkins -y
service jenkins start
cat /var/lib/jenkins/secrets/initialAdminPassword
#chmod 777 jenkins.sh
./jenkins.sh
```

Copy the link

<https://raw.githubusercontent.com/akshu20791/Deployment-script/main/jenkins.sh>

```
security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [18.5 kB]
security.ubuntu.com/ubuntu noble-security/multiverse Translation-en [4288 B]
security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [208 B]
security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [380 B]
B in 8s (4800 kB/s)
e lists... Done
dency tree... Done
nformation... Done
can be upgraded. Run 'apt list --upgradable' to see them.
-32-27:/home/ubuntu# wget https://raw.githubusercontent.com/Tejas-kandregula/Deployment-script/refs/heads/main/jenkins.sh
```

c26c0c1b (jenkins)

2.108.45 PrivateIPs: 172.31.32.27

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates.

Priva

```
2025-08-20 08:14:51 (39.6 MB/s) - 'jenkins.sh' saved [642/642]
root@ip-172-31-32-27:/home/ubuntu# chmod 700 jenkins.sh
```

i-01ec83d0dc26c0c1b (jenkins)

PublicIPs: 107.22.108.45 PrivateIPs: 172.31.32.27

CloudShell Feedback

## Running the jenkins file

```
root@ip-172-31-32-27:/home/ubuntu# ./jenkins.sh
```

## Lets install sonarqube file Also

[https://docs.google.com/document/d/1oBao8evD-P6ZQuexpEc\\_8hTVQHfa0Cx/edit?rtpof=true&tab=t.0](https://docs.google.com/document/d/1oBao8evD-P6ZQuexpEc_8hTVQHfa0Cx/edit?rtpof=true&tab=t.0)

```
Building dependency tree... Done
Reading state information... Done
109 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ip-172-31-91-147:/home/ubuntu# sudo apt update -y && sudo apt upgrade -y
sudo apt install -y docker.io docker-compose unzip git
sudo systemctl start docker
sudo systemctl enable docker
```

i-049b45cf198d41129 (sonarcube)

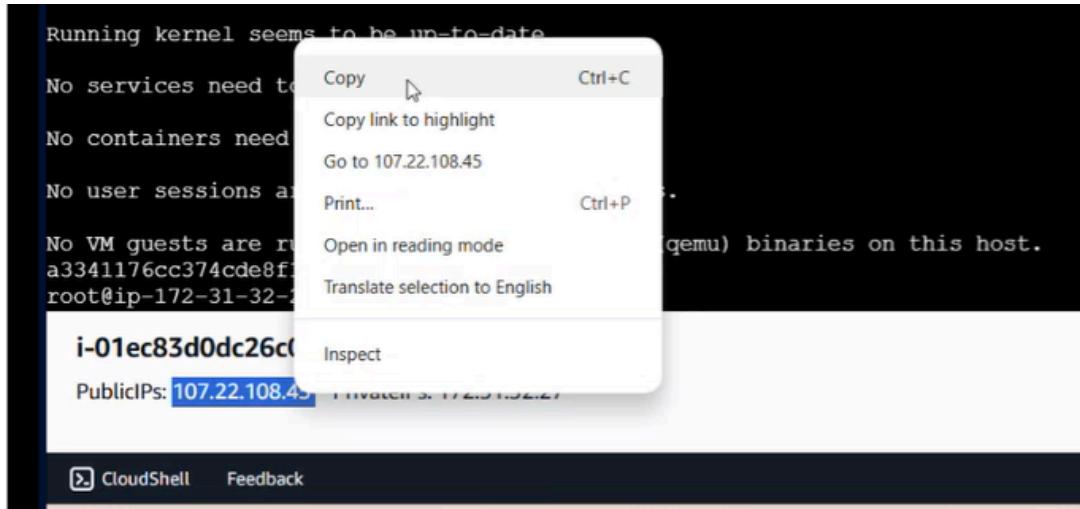
PublicIPs: 3.82.120.131 PrivateIPs: 172.31.91.147

*sudo apt update -y && sudo apt upgrade -y  
sudo apt install -y docker.io docker-compose unzip git  
sudo systemctl start docker  
sudo systemctl enable docker*

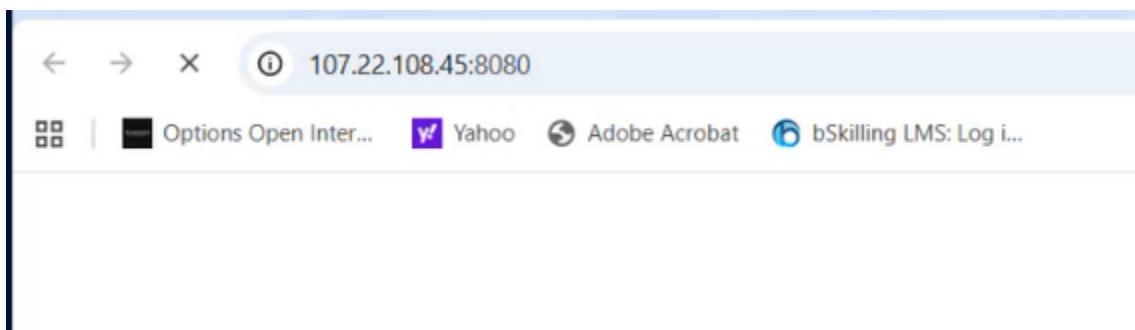
## Got the code to access jenkins file

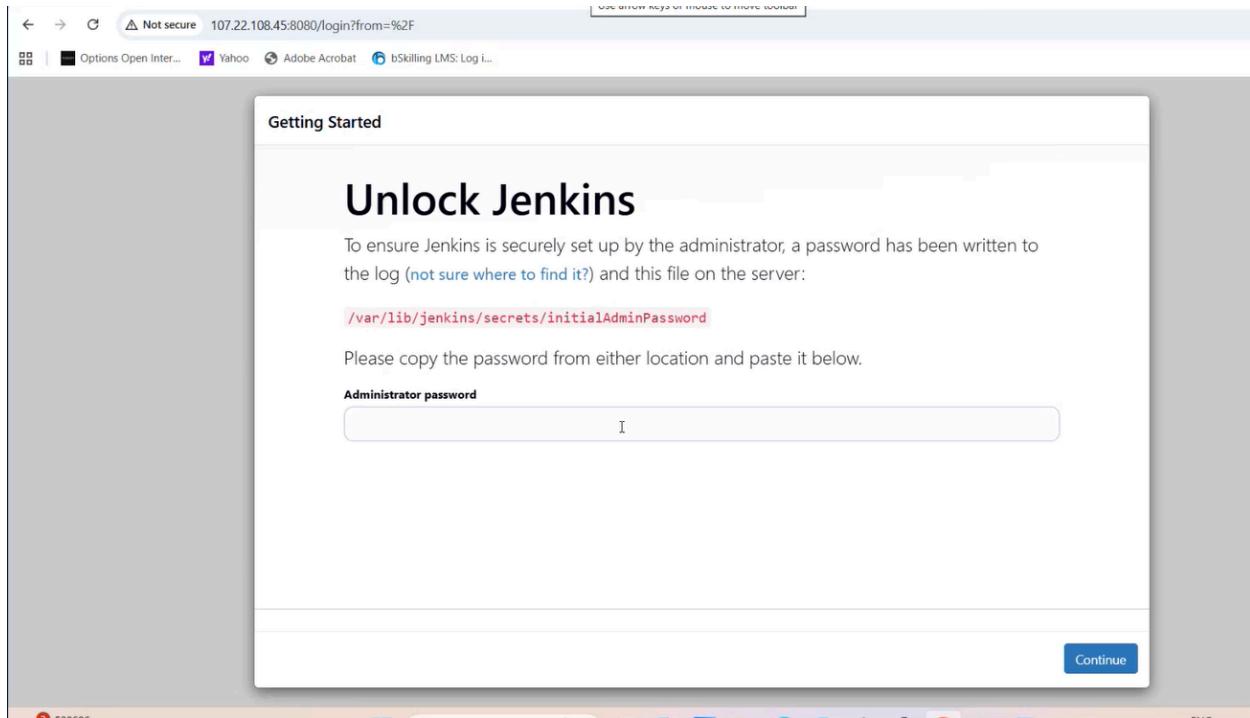
```
No user sessions are running outdated binaries.  
No VM guests are running outdated hypervisor (qemu) binaries on this host.  
a3341176cc374cde8f1d7417a6551d29  
root@ip-172-31-32-27:/home/ubuntu#  
  
i-01ec83d0dc26c0c1b (jenkins)  
PublicIPs: 107.22.108.45 PrivateIPs: 172.31.32.27
```

## Copy the public IP



## Paste it with 8080 port





This is the password

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.  
a3341176cc374cde8f1d7417a6551d29  
root@ip-172-31-32-27:/home/ubuntu#
```

**i-01ec83d0dc26c0c1b (jenkins)**

Public IPs: 107.22.108.45 Private IPs: 172.31.32.27

Getting Started

# Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

## Install suggested plugins

Install plugins the Jenkins community finds most useful.

## Select plugins to install

Select and install plugins most suitable for your needs.

## Now on sonarcube

### 4. Create Docker Compose File

Create a directory and docker-compose.yml file:

```
mkdir sonarqube-docker && cd sonarqube-docker  
nano docker-compose.yml
```

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.  
root@ip-172-31-91-147:/home/ubuntu# mkdir sonarqube-docker && cd sonarqube-docker  
nano docker-compose.yml
```

i-049b45cf198d41129 (sonarcube)

PublicIPs: 3.82.120.131 PrivateIPs: 172.31.91.147

[CloudShell](#) [Feedback](#)

The screenshot shows a terminal window within the AWS CloudShell interface. The terminal title is "aws" and the command being run is "nano docker-compose.yml". The file content is as follows:

```
GNU nano 7.2
environment:
  - POSTGRES_USER=sonar
  - POSTGRES_PASSWORD=sonar
  - POSTGRES_DB=sonarqube
volumes:
  - postgresql:/var/lib/postgresql/data

volumes:
  sonarqube_data:
  sonarqube_extensions:
  postgresql:
I
```

At the bottom of the terminal, there is a key mapping legend and some system information:

**Key Mapping Legend:**

- G** Help
- O** Write Out
- W** Where Is
- K** Cut
- T** Execute
- C** Location
- U** Undo
- A** Set
- X** Exit
- R** Read File
- \** Replace
- U** Paste
- J** Justify
- /** Go To Line
- E** Redo
- 6** Cop

**System Information:**

i-049b45cf198d41129 (sonarcube)  
PublicIPs: 3.82.120.131 PrivateIPs: 172.31.91.147

CloudShell Feedback © 2025, Amazon

Paste the whole in nano

The after copying it press ctrl x and y and enter

Run the sonarcube

## 5. Start SonarQube

Run:

*sudo docker-compose up -d*

Paste the following YAML into the file:

```
version: '3'
services:
  sonarqube:
    image: sonarqube:latest
    container_name: sonarqube

depends_on:
  - db
ports:
  - "9000:9000"
environment:
  - SONAR_JDBC_URL=jdbc:postgresql://db:5432/sonarqube
  - SONAR_JDBC_USERNAME=sonar
  - SONAR_JDBC_PASSWORD=sonar
volumes:
  - sonarqube_data:/opt/sonarqube/data
  - sonarqube_extensions:/opt/sonarqube/extensions

db:
  image: postgres:13
  container_name: postgres
  environment:
    - POSTGRES_USER=sonar
    - POSTGRES_PASSWORD=sonar
    - POSTGRES_DB=sonarqube
  volumes:
    - postgresql:/var/lib/postgresql/data

volumes:
  sonarqube_data:
  sonarqube_extensions:
  postgresql:
```

```
5871bbadd7: Pull complete
e4b4015f66e4: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:6d64b6cdec730c81a8ccbf266716f342d5c9d8a8c93426a2377e239d3ac79a38
Status: Downloaded newer image for sonarqube:latest
Creating postgres ... done
Creating sonarqube ... done
root@ip-172-31-91-147:/home/ubuntu/sonarqube-docker#
```

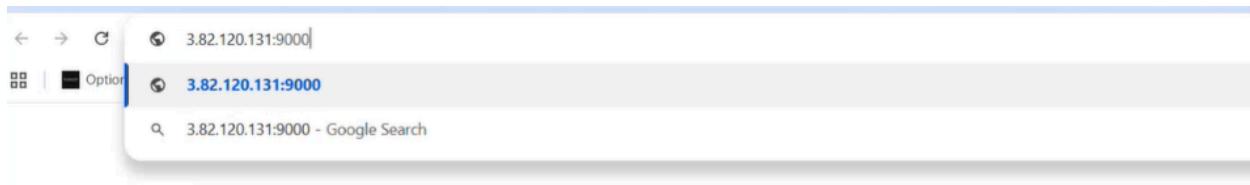
i-049b45cf198d41129 (sonarcube)

PublicIPs: 3.82.120.131 PrivateIPs: 172.31.91.147

 CloudShell  Feedback

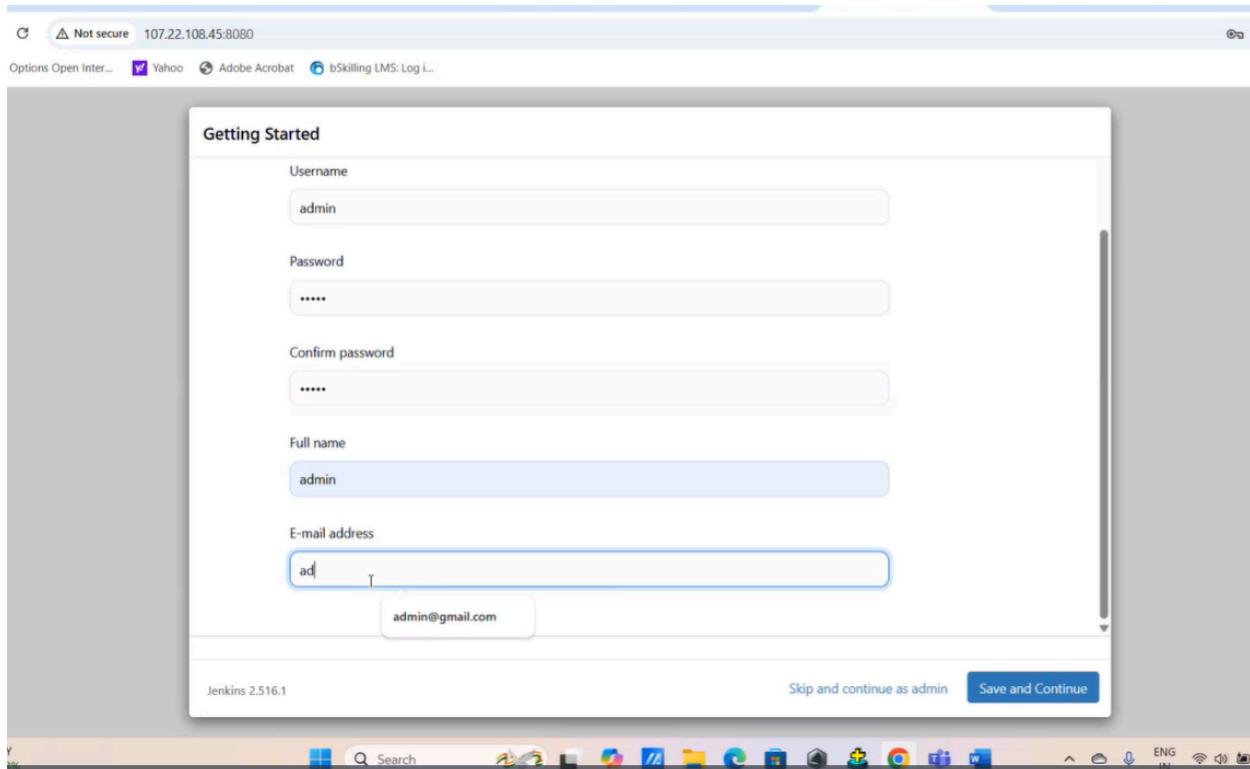
Take the sonarcube public id

Paste it in new tab with ip address 9000



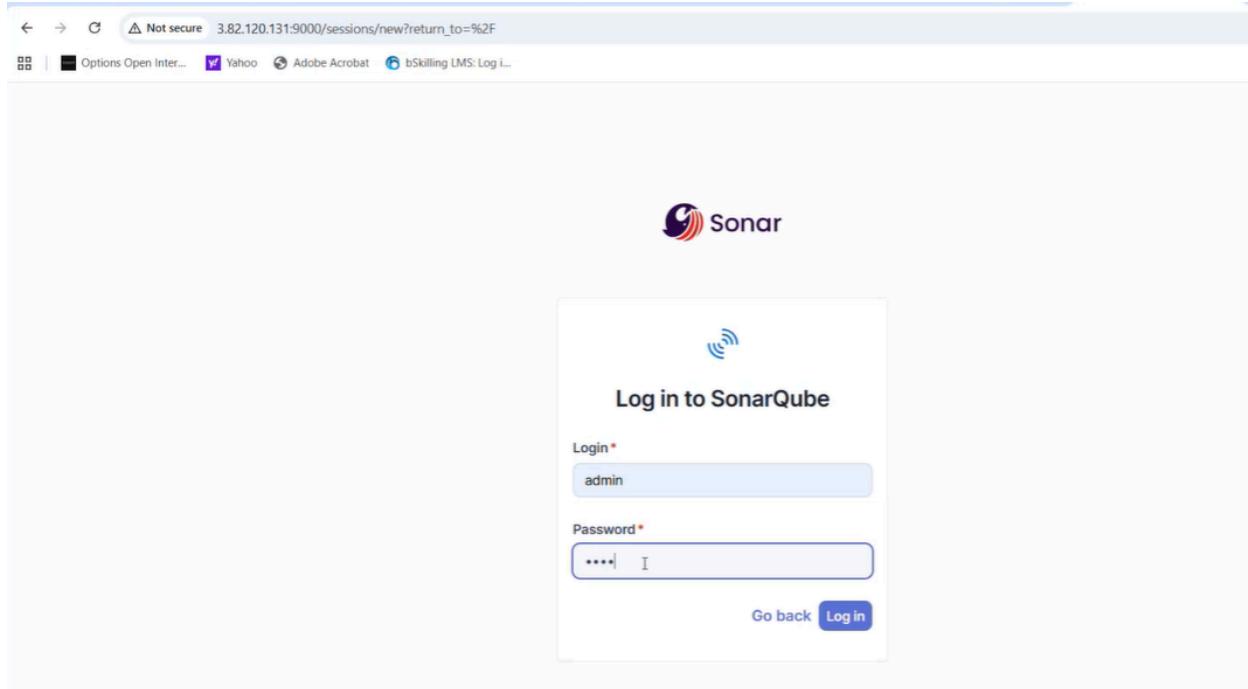
Then come to jenkins

Now came to jenkins

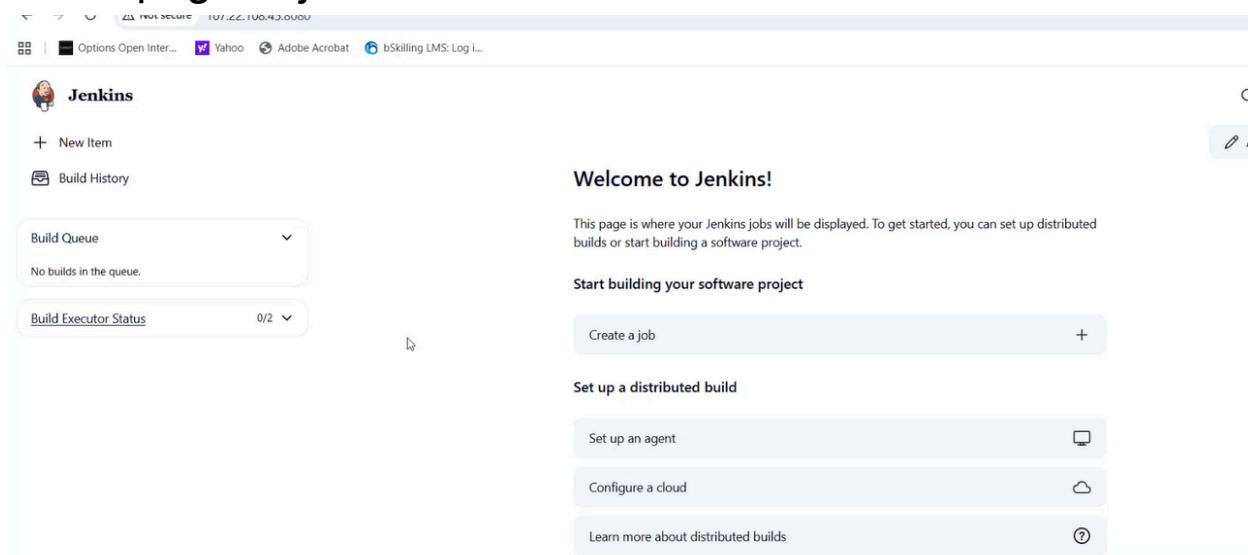


Give all details Admin

Now Login the sonarqube  
Password As (Admin) All things as admin



Home page of jenkins



We have to Integrate Sonarcube with jenkins  
#)Go to Administrator and click on security

← → ⌂ Not secure 3.82.120.131:9000/account/security

≡ Options Open Inter... Yahoo Adobe Acrobat bSkillLMS: Log i...

SonarQube community Projects Issues Rules Quality Profiles Quality Gates Administration More

A Administrator

Profile Security Notifications Projects

Administrator

My Account

Log out

## Security

If you want to enforce security by not providing credentials of a real SonarQube user to run your code scan or to invoke web services, you can provide a User Token as a replacement of the user login. This will increase the security of your installation by not letting your analysis user's password going through your network.

### Generate Tokens

≡ Options Open Inter... Yahoo Adobe Acrobat bSkillLMS: Log i...

SonarQube community Projects Issues Rules Quality Profiles Quality Gates Administration More

A Administrator

Profile Security Notifications Projects

## Security

If you want to enforce security by not providing credentials of a real SonarQube user to run your code scan or to invoke web services, you can provide a User Token as a replacement of the user login. This will increase the security of your installation by not letting your analysis user's password going through your network.

### Generate Tokens

Name	Type	Expires in
Enter Token Name	Select Token Type	30 days

No tokens

Enter a new password

SonarQube community Projects Issues Rules Quality Profiles Quality Gates Administration More

A Administrator

Profile Security Notifications Projects

## Security

If you want to enforce security by not providing credentials of a real SonarQube user to run your code scan or to invoke web services, you can provide a User Token as a replacement of the user login. This will increase the security of your installation by not letting your analysis user's password going through your network.

### Generate Tokens

Name	Type	Expires in
jenkins	Global Analysis Token	30 days

No tokens

community

A Administrator

Profile Security Notifications Projects

## Security

If you want to enforce security by not providing credentials of a real SonarQube user to run your code scan or to invoke web services, you can use tokens instead of the user login. This will increase the security of your installation by not letting your analysis user's password going through your network.

### Generate Tokens

Name	Type	Expires in
Enter Token Name	Select Token Type	30 days

New token "jenkins" has been created. Make sure you copy it now, you won't be able to see it again!

sqa\_6128c13839bbba0327984c727fb88c738eb394af

Name	Day 23 session recording part-2	Last use	Created

## Go to manage jenkins

The screenshot shows the Jenkins dashboard. On the left, there are navigation links for 'New Item' and 'Build History'. In the center, a large 'Welcome to Jenkins!' heading is displayed with the subtext: 'This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.' Below this, there are three main sections: 'Start building your software project' (with a 'Create a job' button), 'Set up a distributed build' (with 'Set up an agent' and 'Configure a cloud' buttons), and a link to 'Learn more about distributed builds'. A 'Manage Jenkins' button with a notification count of '2' is located in the top right corner of the dashboard area.

## Add some plugins required

Not secure 107.22.108.42:8080/manage/

Options Open Inter... Yahoo Adobe Acrobat bSkilling LMS: Log i...

**Jenkins** / Manage Jenkins

Manage Jenkins

Building on the built-in node can be a security issue. You should set up distributed builds. See [the documentation](#).

Set up agent Set up cloud Dismiss

**Java 17 end of life in Jenkins**

You are running Jenkins on Java 17, support for which will end on or after Mar 31, 2026. Refer to [the documentation](#) for more details.

More Info Ignore

System Configuration

- System**  
Configure global settings and paths.
- Tools**  
Configure tools, their locations and automatic installers.
- Nodes**  
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.
- Clouds**  
Add, remove, and configure cloud instances to provision agents on-demand.
- Plugins**  
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
- Appearance**  
Configure the look and feel of Jenkins

Security

107.22.108.45:8080/manage/pluginManager

← → ⌂ Not secure 107.22.108.45:8080/manage/pluginManager/

Options Open Inter... Yahoo Adobe Acrobat bSkilling LMS: Log i...

**Jenkins** / Manage Jenkins / Plugins

**Plugins**

Search plugin updates

Updates Available plugins Installed plugins Advanced settings Download progress

No updates available

Disabled rows are already upgraded, awaiting restart. Shaded but selectable rows are in progress or failed.

Jenkins / Manage Jenkins / Plugins

Plugins

Updates Available plugins **Installed plugins** Advanced settings Download progress

Search: mav

Install	Name	Version
<input type="checkbox"/>	PAM Authentication	1.12
	Security	
	Adds Unix Pluggable Authentication Module (PAM) support to Jenkins	
<b>Day 23 session recording part-2</b>		
<input type="checkbox"/>	JavaMail API	1.6.2-11
	Library plugins (for use by other plugins)	
	This plugin provides the JavaMail API for other plugins.	
<input type="checkbox"/>	Command Agent Launcher	123.v37cfdc92ef67
	Agent Management	
	Allows agents to be launched using a specified command.	
<input type="checkbox"/>	Oracle Java SE Development Kit Installer	83.v417146707a_3d
	Allows the Oracle Java SE Development Kit (JDK) to be installed via download.	
<input type="checkbox"/>	SSH server	3.374.v19b_d59ce6610
	Adds SSH server functionality to Jenkins, exposing CLI commands through it.	

Jenkins / Manage Jenkins / Plugins

Plugins

Updates Available plugins **Installed plugins** Advanced settings Download progress

Search plugin updates

Disabled rows are already upgraded, awaiting restart. Shaded but self

Enter Installed plugins and Enter maven

The screenshot shows the Jenkins plugin manager interface. A search bar at the top contains the text 'maven'. Below it, a sidebar on the left has tabs for 'Updates', 'Available plugins' (which is selected), 'Installed plugins', 'Advanced settings', and 'Download progress'. The main area displays a list of Maven-related plugins:

- Maven Integration 3.27** (Build Tools) - Released 8 days 12 hr ago
- Config File Provider** 994.v3d4a\_5fa\_f353a\_ (Groovy-related, External Site/Tool Integrations, Maven) - Published 11 days ago
- Jira 3.18** (External Site/Tool Integrations, Maven, jira) - Published 1 mo 23 days ago
- Pipeline Maven Integration** 1563.v2e75645a\_7d43 (pipeline, Maven) - Published 8 days 23 hr ago

Click on maven integration  
Visualize the result OWASP Dependency-Check  
Check Out On SonarCube Scanner  
Check out in Soanar Quality Check

The screenshot shows the Jenkins plugin manager interface with a search bar containing 'owas'. The sidebar on the left is identical to the previous screenshot. The main area displays a list of OWASP-related plugins:

- Maven Integration 3.27** (Build Tools) - Released 8 days 12 hr ago (status 100)
- OWASP Dependency-Check 5.6.1** (Security, DevOps, Build Tools, Build Reports) - Published 4 mo 1 day ago (status 100)
- OWASP Dependency-Track 6.0.1** (pipeline, Security, Build Tools, Build Reports) - Published 4 mo 29 days ago (status 99)
- Official OWASP ZAP 1.1.0** (The Official OWASP ZAP Jenkins Plugin) - Published (status 99)

The screenshot shows the Jenkins Plugins management interface. On the left, there's a sidebar with links for Updates, Available plugins (which is selected), Installed plugins, Advanced settings, and Download progress. The main area has a search bar at the top with the query 'sonar'. Below it is a table with columns for Install, Name, Released, and Health. Four plugins are listed:

Install	Name	Released	Health
<input checked="" type="checkbox"/>	Maven Integration 3.27 Build Tools	8 days 12 hr ago	(100)
<input checked="" type="checkbox"/>	OWASP Dependency-Check 5.6.1 Security DevOps Build Tools Build Reports	4 mo 1 day ago	(100)
<input checked="" type="checkbox"/>	SonarQube Scanner 2.18 External Site/Tool Integrations Build Reports	6 mo 23 days ago	(88)
<input checked="" type="checkbox"/>	Sonar Quality Gates 352.vdcdb_d7994fb_6 Library plugins (for use by other plugins) analysis Other Post-Build Actions	5 mo 28 days ago	(100)

Click on Install  
Click on Go Back to top page

The screenshot shows the Jenkins Plugins management interface. The sidebar is identical to the previous one. The main area lists the currently installed plugins with their status:

Plugin	Status
Email Extension	Success
Mailer	Success
Theme Manager	Success
Dark Theme	Success
Loading plugin extensions	Success
Dev Tools Symbols API	Success
Javadoc	Success
JSch dependency	Success
Maven Integration	Success
OWASP Dependency-Check	Success
SonarQube Scanner	Success
Apache HttpComponents Client 5.x API	Success
Sonar Quality Gates	Success
Loading plugin extensions	Success

Below the list, there are two buttons:

- [Go back to the top page](#)  
(you can start using the installed plugins right away)
- Restart Jenkins when installation is complete and no jobs are running

Go to manage Jenkins

distributed

**Go to tools**

**Jenkins / Manage Jenkins**

**Java 17 end of life in Jenkins**

You are running Jenkins on Java 17, support for which will end on or after Mar 31, 2026. Refer to the [documentation](#) for more details.

**More Info**

**System Configuration**

- System** Configure global settings and paths.
- Tools** Configure tools, their locations and automatic installers.
- Plugins** Add, remove, disable or enable plugins that can enhance functionality of Jenkins.
- Nodes** Add, remove, control and monitor the various nodes that Jenkins runs jobs on.
- Clouds** Add, remove, and configure cloud instances to provision agents on-demand.
- Appearance** Configure the look and feel of Jenkins

**Security**

- Security** Secure Jenkins; define who is allowed to access/use the system.
- Credentials** Configure credentials
- Credential Providers** Configure the credential providers and types

[107.22.108.45:8080/manage/configureTools](http://107.22.108.45:8080/manage/configureTools)

**Jenkins / Manage Jenkins / Tools**

**SonarQube Scanner installations**

Add SonarQube Scanner

**SonarQube Scanner**

Name: sonarqube Required

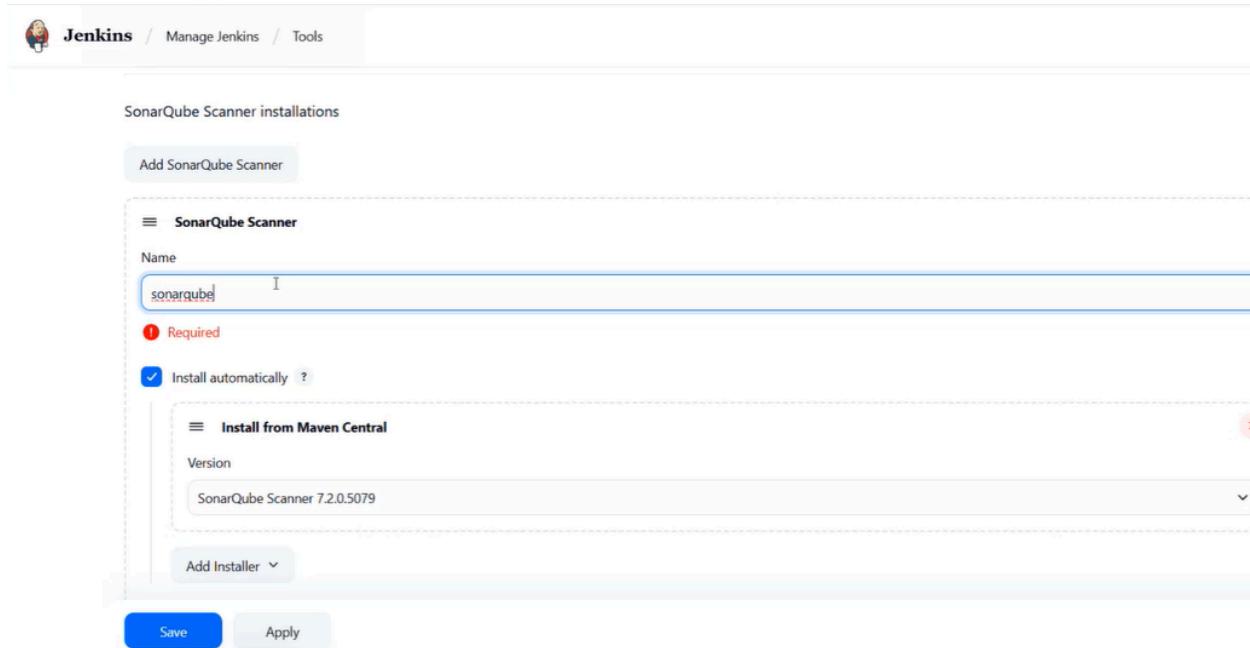
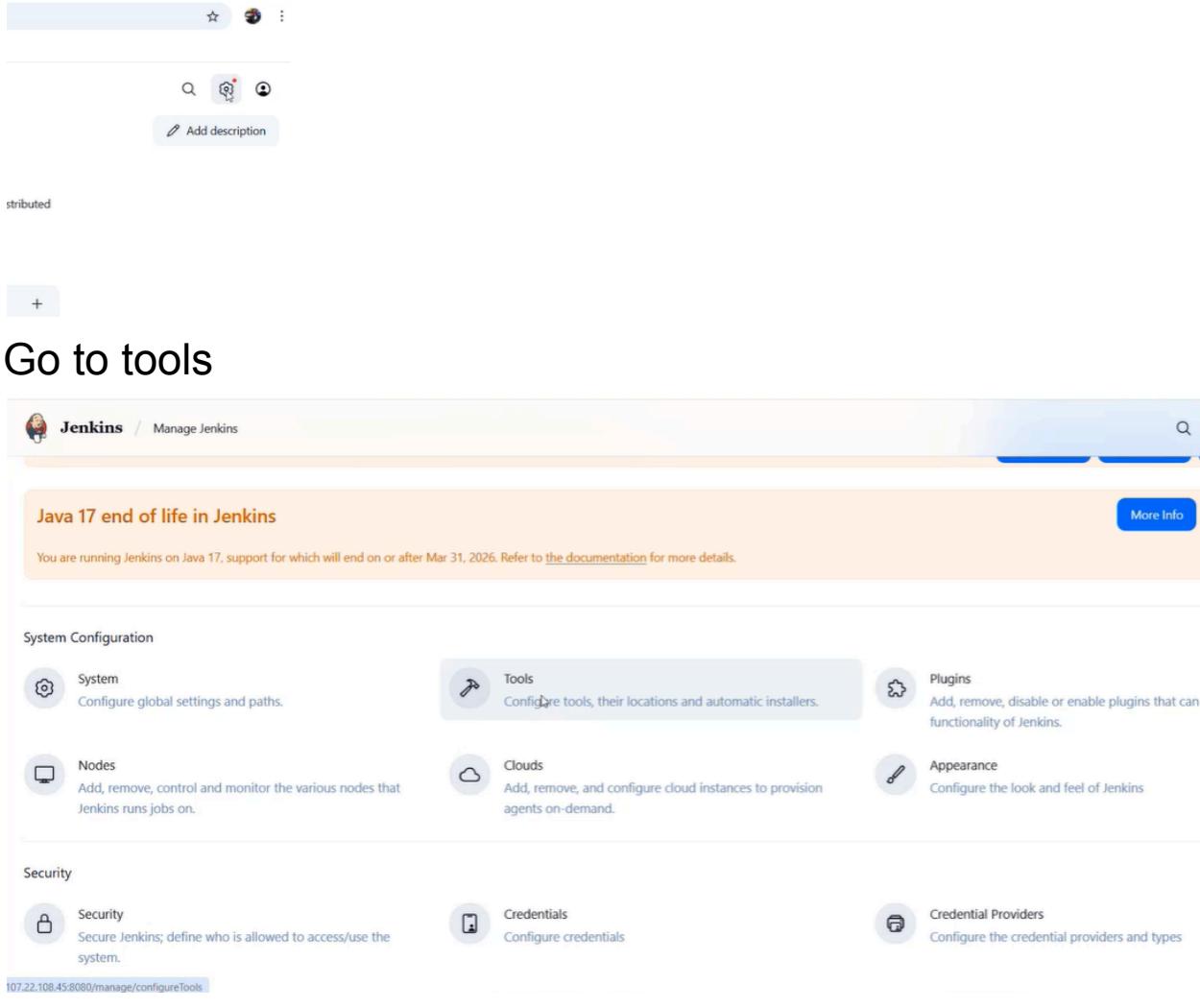
Install automatically ?

**Install from Maven Central**

Version: SonarQube Scanner 7.2.0.5079

Add Installer

Save Apply



# Name it As Sonarcube

The screenshot shows the Jenkins Manage Jenkins interface under the 'Tools' section. It displays two sections: 'Ant installations' and 'Maven installations'. The 'Maven installations' section is expanded, showing a form to add a new Maven installation. The form has a 'Name' field, a note that it is 'Required', and a checked checkbox for 'Install automatically'. Below the form is a link to 'Install from Apache'. At the bottom of the section are 'Save' and 'Apply' buttons.

Write name As Maven and Click On Apply

GO To Again Manage Jenkins  
Go to System

The screenshot shows the Jenkins Manage Jenkins interface under the 'System Configuration' section. It features several configuration items: 'System' (Configure global settings and paths), 'Nodes' (Add, remove, control and monitor the various nodes that Jenkins runs jobs on), 'Tools' (Configure tools, their locations and automatic installers), 'Clouds' (Add, remove, and configure cloud instances to provision agents on-demand), 'Plugins' (Add, remove, disable or enable functionality of Jenkins), and 'Appearance' (Configure the look and feel). A prominent orange banner at the top states 'Java 17 end of life in Jenkins' and provides a link to documentation. A 'Set up agent' button is also visible.

 Jenkins / Manage Jenkins / System

Without a resource root URL, resources will be served from the Jenkins URL with Content-Security-Policy set.

**Global properties**

Disable deferred wipeout on this node ?

Disk Space Monitoring Thresholds

Environment variables

Tool Locations

---

**SonarQube servers**

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

Environment variables

**SonarQube installations**

List of SonarQube installations

Add SonarQube

**Save** **Apply**

 Jenkins / Manage Jenkins / System

Environment variables

Tool Locations

---

**SonarQube servers**

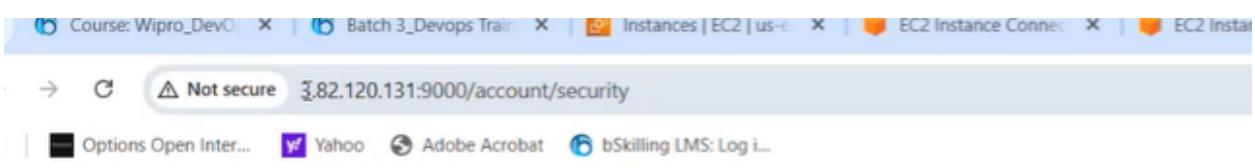
If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

Environment variables

**SonarQube installations**

List of SonarQube installations

Add SonarQube





Name

sonarqube

Server URL

Default is <http://localhost:9000>

http://3.82.120.131:9000

Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled.

- none -



+ Add

Advanced ▾

Add SonarQube

## Click on Apply and save

- Show the entire commit summary in changes ?
- Hide credential usage in job output ?
- Disable performance enhancements ?
- Preserve second fetch during checkout ?
- Add git tag action to jobs ?

Saved

Shell

**Manage Jenkins**

Building on the built-in node can be a security issue. You should set up distributed builds. See [the documentation](#).

**Java 17 end of life in Jenkins**

You are running Jenkins on Java 17, support for which will end on or after Mar 31, 2026. Refer to [the documentation](#) for more details.

**System Configuration**

- System**: Configure global settings and paths.
- Tools**: Configure tools, their locations and automatic installers. (Selected)
- Nodes**: Add, remove, control and monitor the various nodes that Jenkins runs jobs on.
- Clouds**: Add, remove, and configure cloud instances to provision agents on-demand.
- Plugins**: Add, remove, disable or enable plugins that can extend functionality of Jenkins.
- Appearance**: Configure the look and feel of Jenkins.

#### Security

## Click maven jenkins

**Leave site?**  
Changes you made may not be saved.

**Leave** **Cancel**

**Gradle installations**

**SonarScanner for MSBuild installations**

## Go to system

**System**

A long list of system configurations, many of which are redacted with gray bars.

SonarQube community

Projects Issues Rules Quality Profiles Quality Gates Administration More

A Administrator

Profile Security Notifications Projects

### Generate Tokens

Name	Type	Expires in
jenkin	Global Analysis Token	30 days

Name	Type	Project	Last use	Created	Expiration
No tokens					

Enter a new password

Old Password \*

## Copy the token

### Generate Tokens

Name	Type	Expires in
Enter Token Name	Select Token Type	30 days

New token "jenkin" has been created. Copy it now, you won't be able to see it again!

sqa\_5dfaff66f5b7e793cf7d0d456174a9350056f4e7 

Go to manage credentials and select System  
Then Click oN System

Jenkins / Manage Jenkins

Nodes Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

Clouds Add, remove, and configure cloud instances to provision agents on-demand.

Appearance Configure the look and feel of Jenkins

Security Secure Jenkins; define who is allowed to access/use the system.

Credentials Configure credentials 

Credential Providers Configure the credential providers and types

Users Create/delete/modify users that can log in to this Jenkins.

Status Information

The screenshot shows the Jenkins 'Credentials' page. At the top, there are links for 'Manage Jenkins' and 'Credentials'. Below this, the title 'Credentials' is displayed. A table header row includes columns for 'T', 'P', 'Store ↓', 'Domain', 'ID', and 'Name'. Under the heading 'Stores scoped to Jenkins', there is a table with one row. This row has columns for 'P', 'Store ↓', 'Domains', 'Icon', and 'Name'. The 'Domains' column shows '(global)'. The 'Icon' column contains a small Jenkins logo icon. The 'Name' column is empty. At the bottom of the table, there are filter buttons for 'Icon: S M L' and a search bar.

## Click On Global Credentials

The screenshot shows the Jenkins 'Global credentials (unrestricted)' page. The URL in the address bar is 'Jenkins / Manage Jenkins / Credentials / System / Global credentials (unrestricted)'. The page title is 'Global credentials (unrestricted)'. It displays a message: 'Credentials that should be available irrespective of domain specification to requirements matching.' Below this is a table with columns 'ID', 'Name', 'Kind', and 'Description'. A message in the center of the table says 'This credential domain is empty. How about [adding some credentials?](#)'. At the bottom, there are filter buttons for 'Icon: S M L' and a search bar. In the top right corner, there is a blue button labeled '+ Add Credentials' with a cursor pointing at it.

Jenkins / Manage Jenkins / Credentials / System / Global credentials (unrestricted)

### New credentials

Kind: Secret text

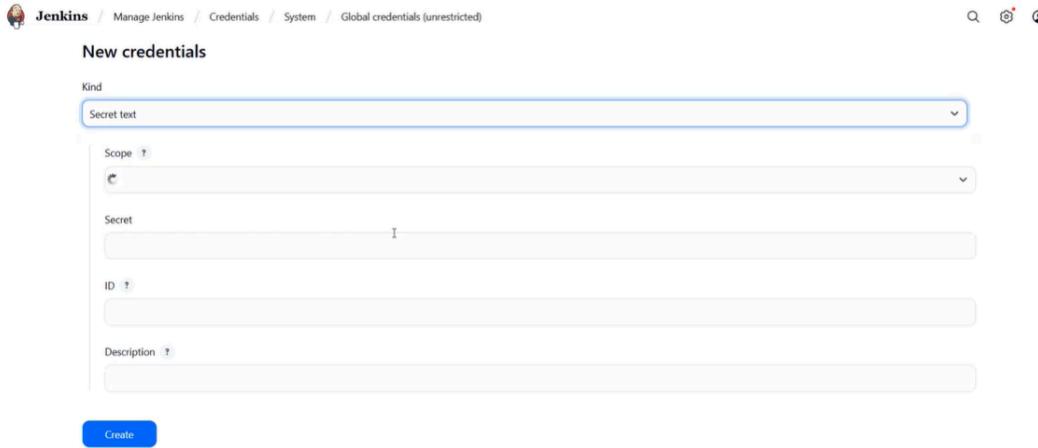
Scope: Global (Jenkins, nodes, items, all child items, etc)

Secret: (Input field)

ID: (Input field)

Description: (Input field)

**Create**



## Copy the sonarcube Token Here

Jenkins / Manage Jenkins / Credentials / System / Global credentials (unrestricted)

Kind: Secret text

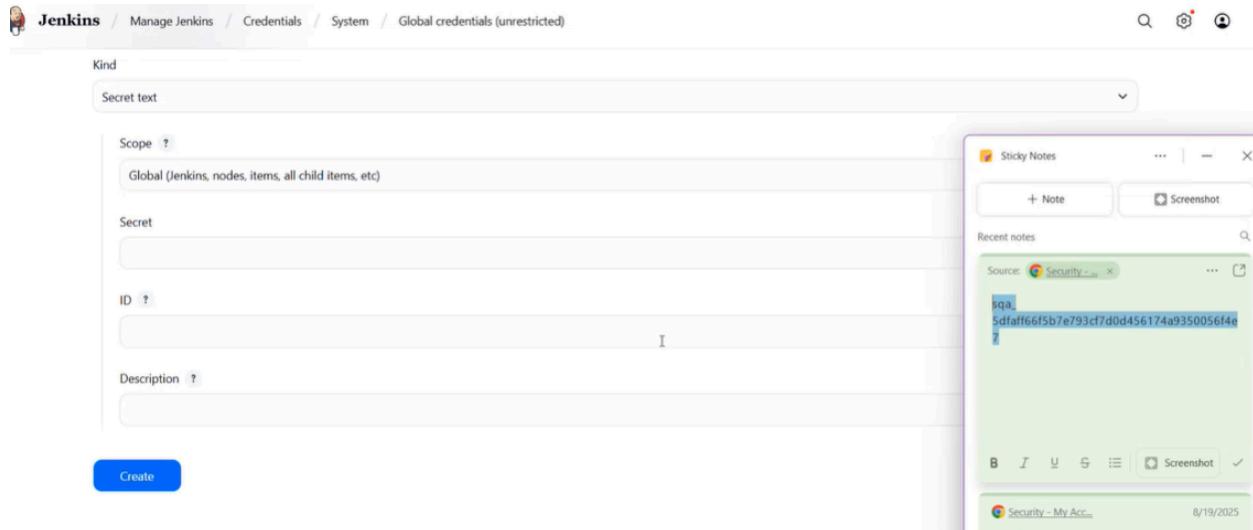
Scope: Global (Jenkins, nodes, items, all child items, etc)

Secret: (Input field)

ID: (Input field)

Description: (Input field)

**Create**



A sticky notes application window is overlaid on the Jenkins interface. It shows a note with the text "sqal 5d1aff66f5b7e793cf7d0d456174a9350056f4e". The note has a "Source: Security - My Acc..." link and a timestamp "8/19/2025".

## Take All Things As SonarCube and then Click it Then Again Go to maven Jenkins

The screenshot shows the Jenkins Global credentials page. The URL is [Manage Jenkins / Credentials / System / Global credentials \(unrestricted\)](#). On the left, a sidebar lists various Jenkins management options like System Configuration, Tools, Plugins, Nodes, Clouds, Appearance, Security, Credentials, and Credential Providers. The main content area displays a table of global credentials:

Name	Kind
sonarcube	Secret text

## Then Again Go to System

The screenshot shows the Jenkins System configuration page for SonarQube. The URL is [Manage Jenkins / System](#). The page has fields for Name (sonarqube), Server URL (http://3.82.120.131:9000), and Server authentication token (sonarcube, selected from a dropdown menu). At the bottom are Save and Apply buttons.

We will Find Sonarcube here  
Click on apply and save  
Now Click on New Item

The screenshot shows the Jenkins dashboard. At the top left is the Jenkins logo and the word "Jenkins". Below it are two buttons: "+ New Item" (highlighted with a mouse cursor) and "Build History". Underneath is a "Build Queue" section with a dropdown menu set to "No builds in the queue.". To the right is a "Welcome to Jenkins!" message: "This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project." Below this is a "Start building your software project" section with a "Create a job" button. Further down is a "Set up a distributed build" section with three buttons: "Set up an agent", "Configure a cloud", and "Learn more about distributed builds".

The screenshot shows the "New Item" creation dialog. At the top is the title "New Item". Below it is a text input field with the placeholder "Enter an item name" containing the text "addressbook". A dropdown menu below the input field shows the selected item "addressbook". The list of project types includes:

- Freestyle project**: Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.
- Maven project**: Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.
- Pipeline**: Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
- Multi-configuration project**: Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.
- Folder**: A folder icon.

At the bottom of the dialog is an "OK" button.

Go To pipeline and click on Okay

Jenkins / All / New Item

### New Item

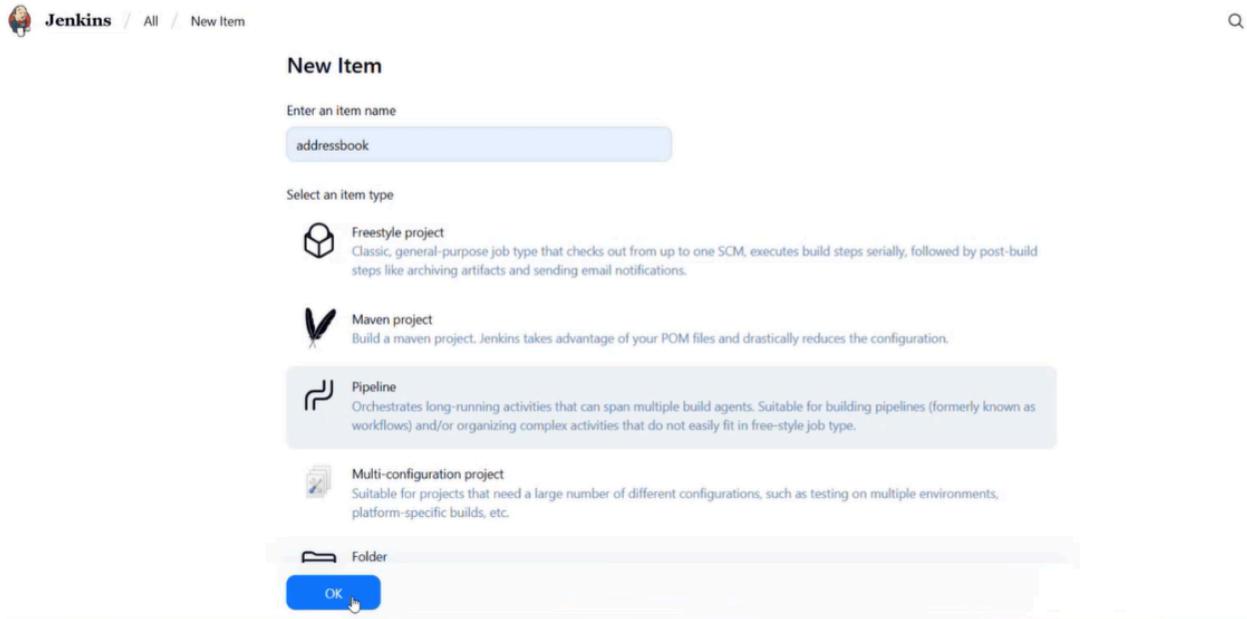
Enter an item name  
addressbook

Select an item type

-  Freestyle project  
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.
-  Maven project  
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.
-  Pipeline  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
-  Multi-configuration project  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder

OK



## Go To pipeLine

Jenkins / addressbook / Configuration

### Configure

General Triggers Pipeline Advanced

### Pipeline

Define your Pipeline using Groovy directly or pull it from source control.

Definition

Pipeline script

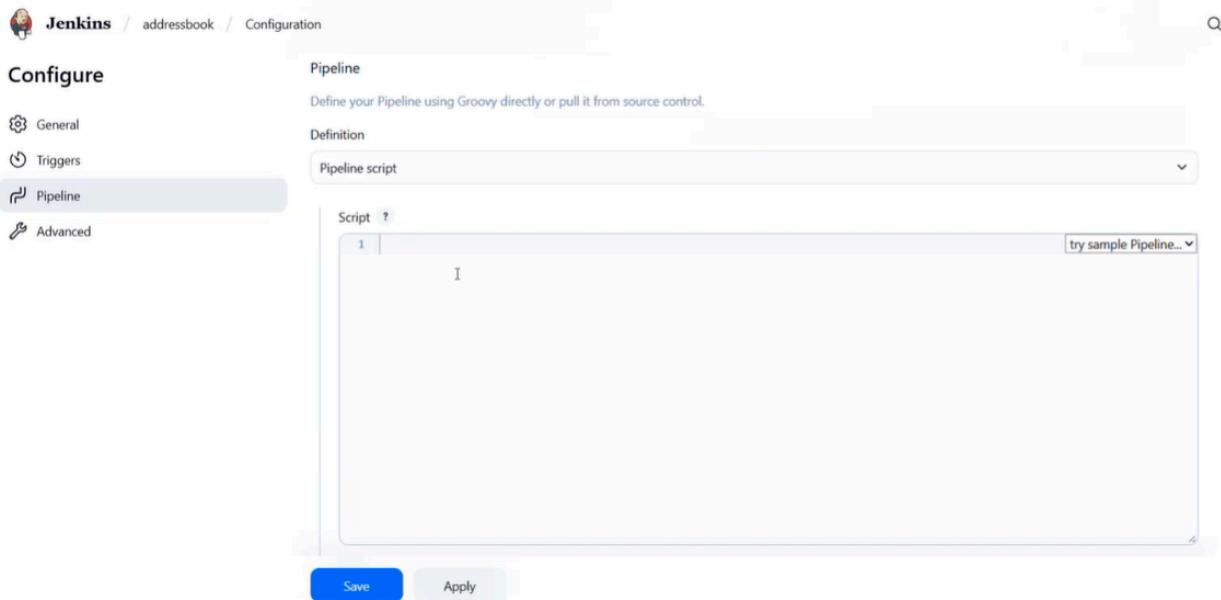
Script ?

```
1
```

I

try sample Pipeline... ▾

Save Apply



Copy the pipeline code and paste t

The screenshot shows the Jenkins Pipeline configuration page. The left sidebar has tabs for General, Triggers, Pipeline (which is selected), and Advanced. The main area is titled 'Pipeline' with the sub-section 'Definition'. A 'Pipeline script' tab is open, showing the following Groovy code:

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38

```

The code defines a pipeline with a stage for SonarQube analysis, setting environment variables for host URL and auth token, and running a Maven command to analyze the project.

We can use our own GitHub repo

```

3
4
5
6
7
8
9
10
11
12
13
14
15

```

The code snippet shows a Jenkins pipeline definition with stages for Maven, Checkout Code (using a GitHub repository URL), and Build.

Replace this url

```

22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38

```

The code snippet shows a Jenkins pipeline definition with stages for Maven test, SonarQube Analysis (using a placeholder URL), and another SonarQube Analysis stage with its own environment and steps.

Cut the old one and copy your own url

```
22          sh 'mvn test'
23      }
24  }
25
26  stage('SonarQube Analysis') {
27      environment {
28          SONAR_HOST_URL = 'http://http://3.82.120.131:9000' // Replace with your SonarQube URL
29          SONAR_AUTH_TOKEN = credentials('sonarcube') // Store your token in Jenkins credentials
30      }
31      steps {
32          sh 'mvn sonar:sonar -Dsonar.projectKey=sample_project -Dsonar.host.url=$SONAR_HOST_URL -Dsonar.login=$SONAR_AUTH_TOKEN'
33      }
--
```

And Click on apply & save  
Click on build Now

The screenshot shows the Jenkins pipeline configuration page for the 'addressbook' pipeline. The pipeline script is displayed at the top, showing a Jenkinsfile with stages for testing and SonarQube analysis. Below the script, the 'Build Now' button is highlighted with a cursor, indicating it has been clicked. The pipeline status is shown as 'addressbook' with a green 'Running' status. The sidebar on the left lists pipeline management options: Status, Changes, Build Now, Configure, Delete Pipeline, Stages, Rename, and Pipeline Syntax.

Jenkins / addressbook

Status Changes Build Now Configure Delete Pipeline Stages Rename Pipeline Syntax

Builds

Today #1 8:36 am

This screenshot shows the Jenkins pipeline status page for the 'addressbook' pipeline. The top navigation bar includes links for Status, Changes, Build Now, Configure, Delete Pipeline, Stages, Rename, and Pipeline Syntax. Below this is a 'Builds' section for today, showing a single build labeled '#1' started at 8:36 am. A progress bar indicates the build is currently running.

Jenkins / addressbook / #2

Status Changes Console Output Edit Build Information Delete build '#2' Timings Git Build Data Pipeline Overview Restart from Stage Replay Pipeline Steps Workspaces Previous Build

#2 (20-Aug-2025, 8:37:44 am)

Started by user admin

This run spent:

- 14 ms waiting;
- 34 sec build duration;
- 34 sec total from scheduled to completion.

git Revision: 74b030bcc2387c964ee4197ef85c6712e79cbca0  
Repository: <https://github.com/Tejas-kandregula/addressbook-cicd-project>  
refs/remotes/origin/master

</> No changes.

This screenshot shows the Jenkins pipeline details for build #2. The top navigation bar includes links for Status, Changes, Console Output, Edit Build Information, Delete build '#2', Timings, Git Build Data, Pipeline Overview, Restart from Stage, Replay, Pipeline Steps, Workspaces, and Previous Build. The main area displays the build number (#2), start time (20-Aug-2025, 8:37:44 am), and a green checkmark indicating success. It also shows the user who started the build (admin) and the time spent. Below this, it provides git revision and repository information. A note at the bottom states 'No changes.'

The way in which security, reliability, and maintainability counts and ratings are calculated has changed. [Learn more in SonarQube documentation](#).

**SonarQube community** Projects Issues Rules Quality Profiles Quality Gates Administration More ▾

My Favorites All

Search projects (minimum 2 characters)

Perspective Overall Status Sort by Name

**Filters**

**Quality Gate**

<input checked="" type="checkbox"/> Passed	0
<input type="checkbox"/> Failed	0

**Security**

<span style="color: green;">A</span> ≥ 0 info issues	0
<span style="color: yellow;">B</span> ≥ 1 low issue	0
<span style="color: orange;">C</span> ≥ 1 medium issue	0
<span style="color: red;">D</span> ≥ 1 high issue	0
<span style="color: red;">E</span> ≥ 1 blocker issue	0

SonarQube™ technology is powered by [SonarSource SA](#). Community Build v25.8.0.112829 - MQR MODE. LGPL v3. Community Documentation.

**SonarQube community** Projects Issues Rules Quality Profiles Quality Gates Administration More ▾

Vaadin Addressbook example Bind project / main

Overview Issues Security Hotspots Code Measures Activity Project Settings Project Information

**main** 1.3k Lines of Code · Version 2.0 · Set as homepage Last analysis 41 seconds ago

**Passed** The last analysis has warnings. See details

New Code Overall Code

Security	Reliability	Maintainability
0 Open issues	A 16 Open issues	C 125 Open issues
Accepted issues	Coverage	Duplications
0	0.0%	3.3%

